Dmitry Svyatogorov

System architect, leading design engineer, main linux system adminisrator.
Specialization in the field of solving queuing tasks *(access to any limited resources)*. From problem statement to being released into production.

Birthday: 01.04.1976
E-mail: ds@vo-ix.ru
Phone.: +7(921)958-3253
telegram, whatsapp: +79219583253
discord: dmitry_svyatogorov

Residence: Russia, Saint Petersburg.

Education

1999 — State Electrotechnical Univercity of Saint-Petersburg, Master of Engineering (M.Eng.), Radio-electronic devices development.

Experience

10.2019 - now	IFuture, Ltd.	Main system administrator in the linux administration group.
09.2015 - 10.2019	SPLAT, Ltd.	Main administrator of computer network. (linux)

Working in the interests of an "internal customer". Development and "L3+" software support for international (Northern hemisphere) data center networks.

Main projects:

- Refinement of the functionality of the deployed "kilo" *(qemu-kvm+libvirt, python 2.7)*. Shared LVM paired with the "usual" one, with slightly fixed "sanlock". Wrapping the management part into chroot-container to manage of actual hypervisors *(Oracle linux 8+)*, attachment of the practically unlimited number of disks *(26^2 and more, from SAN)*. Bug fixing of the original code *(huge pages, vlan mapping in openvswitch)*. Reverse-engineering of the DB to make possible import/export/config recovery and seamless migration of hypervisor among different controller clusters.
- Disaster-resistant multitenant DNS *(my dnsd+execd on GO, transport via kafka, knot DNS on frontends)*. Expanding RFC-2136 with the ability of creation and destruction of domains, and the auto-binding to the tenant.
- The pipeline of "probative" deploy for acquiring services *(passed the first audit according to PCI-DSS 2.0, the service team is now successfully passing all the recurrent audits)*. ZFS+overlayFS+selinux, xen+flask on the production hypervisors. PostgreSQL to work with the checksums (because I know it well enough and it was the shortest way). The pipeline is managed with the maid earlier "schema" modular translator. *(The circuitry is in very general the analog of «kubernetes».)*
- The alerting system about of ip-address duplicates in Ethernet networks *(telemetry agent "ipaddrd" on C, "ipaddr-collector" server on GO and processor on bash)*.
- The service for outsorcing the management of resources supplied from the storage *(CLI on python 3.6, modular topology like yum|dnf, full-featured autocompletion by means of bash-completion)*.
- The keybord layout switcher for linux "xswitcher" *(in times of covid-19 my department was working 4 days in the week, it was time to do something for yourself)*. A practical proof that the user's actions with the keyboard can be fully described through a finite state machine *("regular grammar")*. GO with small bindings to C++ (Xorg).
- Deep refactoring of VX-PBX in the interests of the internal customer. Upgrade to the actual Oracle linux 8, Asterisk 20, PostgreSQL 15. Fixed working with labels in virt. PBX, a new "strict" isolation mode has been implemented *(like one in Eltex unfortunate softswitches)*. Deprecated "sip" was

replaced with "pjsip", while saving full compatibility at the configuration level. Transfer of CDR’s via kafka to Oracle DB *(new GO-based services)*.

- The rule replicator for Normation Rudder *(the French-written Scala control software for cfengine)* with three levels of abstraction reduction.

04.2014 - 09.2015	"Petersburg's Telephone", Ltd.	Leading engineer.

Administration of an electric communication operator (transit and subscriber telephony, subscription Internet).

- Development of a modular translator "schema" *(perl 5, not to be confused with "scheme" for LISP). In order to radically lower the threshold for entry into telephony (for students)*. The ideological analogue of the language "Dragon". *(A similar approach is also used in LAN simulators).*
- Improvements to Asterisk queues for servicing very high loads *(20+ E1 streams = 600+ subscribers at the same time)*.
- *Development of one of the first mass calling systems (yes, there was such a thing…)*.
- Development of an geoIP-based anti-fraud system. *(It did not fit at the operator level, because it reduces income. The subscriber must first spend the money.)*
- Development of circuitry with increased reliability for medical institutions.
- Development of a FreeSWITCH-based front-end to Asterisk to support multiple SIP registrations.
- Improvements in the "upsert" and microcode for Asterisk for manipulating transit calls, etc.

11.2013 - 03.2014	Administration of the mining network "StarLabs".

08.2013 - 11.2013	"CFC-Acquiring", Ltd (the contract for 3 months)	System administrator.
05.2013 - 08.2013	"CFC -Internet-Services", Ltd (the contract for 3 mon.)	System administrator.

Administration of the "Money Online" acquirer at the end of its existence.

- Development of the "upsert" translator for VX-PBX.
- Preparing the VX-PBX for commercial operation. Partitioning into virtual PBXes *(by analogy with namespaces in linux)*. Creation of «VOIX TELECOM», Ltd. for the task.

10.2010 - 04.2013	"Miran", Ltd.	Senior system administrator.
02.2010 - 10.2010	"Miran", Ltd.	System administrator.
12.2008 - 02.2010	"Miran", Ltd.	Assistant to the system administrator.

Administration of an electric communication operator B2B (subscriber telephony, subscription Internet). Transfer of all telematics services from freeBSD 5…7 to linux (Debian, CentOS).

- Development of the registry for LIR *(PostgreSQL already had the "inet" type with the necessary address arithmetic)*.
- Development of a system of email gateways with selective ingress graylisting, checking the availability of the address before sending and restrictions on the intensity of mailing. Exim+postgreSQL+perl.
- Development of a NAT router for "packing" subscribers. 1 ip-address → 1 office behind of NAT.
- Development of billing for Internet services *(netflow 5, post-payment for traffic or per channel)*.
- The beginning of the development of PBX based on Asterisk 1.4 and beyond. Managed Code (AEL) in tight coupling with PostgreSQL 8+ *(the whole routing logic)*. I completely discarded the paradigm of routing by prefixes, introduced the concept of "call level" on the basis of the call billing scheme adopted at that time.

10.1999 - 11.2008	(DOS, Windows, 1C etc.)

It is impossible to grow as an engineer without open source. Only as a "hacker" *(IDA, SoftICE/winICE, and all that).*

Operating systems:

*** linux:**
- RH-based — I prefer it for telematics, now I'm using a slightly modified Oracle-8.
- SUSE-based — all personal PCs and laptops on openSUSE *(until they give up rpm)*.
- Debian-based — "the choice of the majority".

*** freeBSD** — I don't use it after ≈2012.
*** solaris** — maintenance of a homegrown storage system in "StarLabs". After that, I used OpenIndiana for backups in VOIX *(until the stabilization of ZFSonLinux)*.
*** On network equipment:** Cisco IOS, Mikrotik RTOS, openWRT.

Hardware architecture:

Basically, the descendants of the IBM PC. Except for specialized equipment.

Programming languages:

C(89/99), GO(1.8…1.23), Perl 5+, Python 2.4…3+, PL/pgSQL, cfengine, PHP 5…7 *(as without it?)*.

◦ It is important to understand here that I am not so much "programming" as "developing a certain scheme". Which I can only imagine in close-up at the beginning of the way. In this mode, "functional" languages are not suitable: I do not know the solution to the problem to take it and encode. A similar story with the "classes" of the OOP. If they are already in the project, then of course I use them.

◦ Of course, I do not keep all the listed PLs in my memory. I recall PL "for the project".
Shell-scenarios: sh, bash, ash/dash, ksh. Recently — bash-only. (*AWK, pcre, strace, etc.,etc.)*

Технологии:

- **Linux kernel** *(minor edits in the Debian/Ubuntu code)*, **initramfs** *(dragging an additional settings from "/proc/cmdline")*, udevd *(scanning hundreds of LUNs under the new rules halts the hypervisor)*, **systemd** *(a set of my units + debug of the boot order)*.
- **Virtualization:** qemu-kvm, qemu-xen, libvirt/GANETI, openStack, Proxmox VE. From reading code to fill in missing documentation (qemu) to major improvements (openStack).
- **Container isolation:** network and disk namespaces in linux, a bit of LXC in PCI-DSS project. I haven't (yet) got into docker/kubernetes deeply.
- **Telephony: Asterisk** *(up to major code improvements, interaction with the Asterisk Team to pull through the necessary edits)*. **FreeSwitch**: basic functionality *(build/install and configure what you need for the project)*. SIP(SIP-T, SIP-I)/RTP, IAX, DAHDI (E1, fxo/fsx).
- **DB:** PostgreSQL *(deep in sql/plpgsql)*, MySQL/MariaDB *(cross replication configuration scenarios AKA «active-active» (no, standby!), bin-logs, reverse engineering to ensure configuration export from OpenStack)*, Oracle DB *(superficial)*.
- **X.500:** OpenLDAP/re-openldap, dirsrv(freeIPA). Replication architecture, ACL, etc.
- **DNS:** Knot *(several thousand of autorotable domains "active-active " (no, standby!))*, bind9 *(active-standby topology distributed among DC with centralized configuration)*, unbound *(so far, only as caching recursors)*. (+Unsuccessful (overall) experiments with PowerDNS and nsd). Your «dnsd» proxy (DNS>>kafka).
- **Exotic on maintenance (deb/rpm):** ucarp *(I fixed the author's student error with non-monotonic clocks and added some functionality for the customer, wrote and covered auxiliary scripts for deb with integration tests)*, network-scripts/network-scripts-extra *(starting from RH-9 I will have to take it on full support, because the network manager knows nothing about complex topologies)*, cobbler, rudder, sanlock ("xsanlock").
- **Networks: Ethernet** (the base of bgp, ospf, vxlan and so on; L2/L3 tunneling, stp and other necessary tricks, queue disciplines, netflow, etc.), **Fibre Channel** (rdac/alua), **PSTN** (E1/T1 signaling). *The ability to approach the issue with wireShark/tShark.*
- **The rest:** SSL/TLS (openssl), HTTP (nginx/apache), data buses (apache kafka/rabbitMQ), DHCP (ISC dhcpd/dnsmasq), SMTP (exim/postfix), IMAP4/POP3 (dovecot).

- **Process modeling.** *(If there is a model, you can very quickly answer «what's wrong» — build an «expert system», or «train AI».)*

Publications:

https://github.com/ds-voix/ https://habr.com/ru/users/PnDx/publications/articles/

* I do not publish business-sensitive parts regardless of the license, but I can use the developments in new projects.

Additional information:

- Good communication skills («ambivert»).
- Attention to detail. The ability to work «realtime» *(that is, "cheat" where necessary, leaving the in-depth proceedings for the next time)*.
- Skills of both working in pairs *(in a threesome, the third is a leader)*, and in a team.
- Lecturing practice (small audiences, online/offline).
- Leadership of a small (7 people) team in the background. I prefer to build a «win-win»-relationship with all my colleagues. I help them to improve their technical level, and I welcome the same in relation to myself.
- I help to build ITIL processes *(despite the fact that I'm too «resultarian»)*.
- *ISO-9000 basics for egress control («to fly and not fall»)*.

Languages:

- **English:** technical ±fluent and a bit of colloquial vocabulary *(communication is mainly with non-native speakers from other countries)*.
- **Spanish:** 10 years of school, in the passive.
- **Chinese:** ≈HSK-3, Chinese gymnasium for a child.